Int. J. Contemp. Math. Sciences, Vol. 7, 2012, no. 35, 1711 - 1725

Data Mining with Linguistic Thresholds

Tzung-Pei Hong

Department of Electrical Engineering
National University of Kaohsiung
Kaohsiung, Taiwan, R.O.C.
tphong@nuk.edu.tw

Ming-Jer Chiang

Institute of Information Engineering
I-Shou University
Kaohsiung, Taiwan, R.O.C.
ndmc893009@yahoo.com.tw

Shyue-Liang Wang

Department of Computer Science
New York Institute of Technology
New York, USA
slwang@nyit.edu

Abstract

Data mining is the process of extracting desirable knowledge or interesting patterns from existing databases for specific purposes. In the past, the minimum supports and minimum confidences were set at numerical values. Linguistic minimum support and minimum confidence values are, however, more natural and understandable for human beings. This paper thus attempts to propose a new mining approach for extracting interesting weighted association rules from transactions, when the parameters needed in the mining process are given in linguistic terms. Items are also evaluated by managers as linguistic terms to reflect their importance, which are then transformed as fuzzy sets of weights. Fuzzy operations including fuzzy ranking are then used to find weighted large itemsets and association rules.

Mathematics Subject Classification: 62-07

Keywords: association rule, data mining, weighted item, fuzzy set, fuzzy ranking

1 Introduction

Knowledge discovery in databases (KDD) has become a process of considerable interest in recent years as the amounts of data in many databases have grown tremendously large. The KDD process generally consists of three phases: pre-processing, data mining and post-processing [10, 14, 15, 26]. Among them, data mining plays a critical role to KDD.

Recently, the fuzzy set theory has been used more and more frequently in intelligent systems because of its simplicity and similarity to human reasoning [35, 36]. The theory has been applied in fields such as manufacturing, engineering, diagnosis, economics, among others [17, 23, 25, 35]. Several fuzzy learning algorithms for inducing rules from given sets of data have been designed and used to good effect with specific domains [5, 7, 13, 16, 18-21, 29-31]. Strategies based on decision trees were proposed in [9, 11-12, 27-29, 32-33], and based on version spaces were proposed in [30]. Fuzzy mining approaches were proposed in [8, 22, 24, 34].

In the past, the minimum support and minimum confidence values were numerical. In this paper, linguistic minimum support and minimum confidence values are given, which are more natural and understandable for human beings. Also, items may have different importance, which is evaluated by managers or experts as linguistic terms. A novel mining algorithm is then proposed to find weighted linguistic association rules from transaction data. It first transforms linguistic weighted items, minimum supports and minimum confidences into fuzzy sets, then filters weighted large itemsets out by fuzzy operations. Weighted association rules with linguistic supports and confidences are then derived from the weighted large itemsets.

The remaining parts of this paper are organized as follows. Several approaches for mining association rules are reviewed in Section 2. Fuzzy sets and operations are introduced in Section 3. The notation used in this paper is defined in Section 4. The proposed weighted data-mining algorithm for linguistic minimum supports and confidences are described in Section 5. An example is given to illustrate the proposed mining algorithm in Section 6. Conclusions and proposal of future work are given in Section 7.

2 Review of Mining Association Rules

Agrawal and his co-workers proposed several mining algorithms based on the concept of large itemsets to find association rules in transaction data [1-4]. They divided the mining process into two phases. In the first phase, candidate itemsets were generated and counted by scanning the transaction data. If the number of an itemset appearing in the transactions was larger than a pre-defined threshold value (called minimum support), the itemset was considered a large itemset. Itemsets containing only one item were processed first. Large itemsets containing only single items were then combined to form candidate itemsets containing two items. This process was repeated until all large itemsets had been found. In the second phase, association rules were induced from the large itemsets found in the first phase. All possible association combinations for each large itemset were formed, and those with calculated confidence values larger than a predefined threshold (called minimum confidence) were output as association rules.

Cai *et al.* then proposed weighted mining to reflect different importance to different items [6]. Each item was attached a numerical weight given by users. Weighted supports and weighted confidences were then defined to determine interesting association rules. Yue *et al.* then extended their concepts to fuzzy item vectors [34].

In this paper, the fuzzy concepts are used to represent item importance, minimum supports and minimum confidences. These parameters are expressed in linguistic terms, which are more natural and understandable for human beings.

3 Review of Related Fuzzy Concepts

Fuzzy set theory was first proposed by Zadeh and Goguen in 1965 [36]. A function called the membership function, $\mu_A(x)$, is defined for mapping a member x to a memebrship degree between 0 to 1. Triangular membership functions are commonly used and can be denoted by $A = (a, b, c)$, where $a \leq b \leq c$ (Figure 1). The abscissa b represents the variable value with the maximal grade of membership value, i.e. $\mu_A(b)$=1; a and c are the lower and upper bounds of the available area. They are used to reflect the fuzziness of the data.

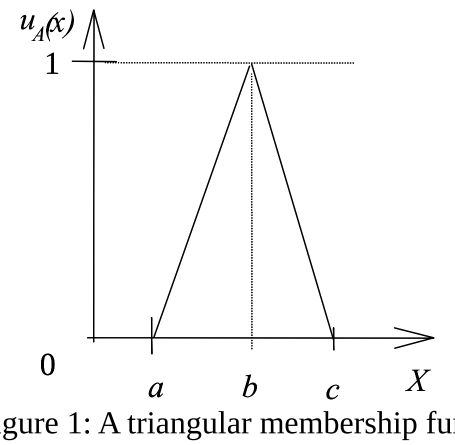


Figure 1: A triangular membership function

There are a variety of fuzzy set operations. Among them, three basic and commonly used operations are *complement*, *union* and *intersection.* Fuzzy ranking is used to determine the order of fuzzy sets and is thus quite important to actual applications. Several fuzzy ranking methods have been proposed in the literature. The ranking method using gravities is used in the paper.

4 Notation

The notation used in this paper is defined as follows.

n: the total number of transaction data;
m: the total number of items;
k: the total number of managers;
A_i: the *i*-th item, $1 \leq i \leq m$;
$W_{i,j}$: the transformed fuzzy weight for importance of item A_i, evaluated by the *j*-th manager;
W_i^{ave}: the fuzzy average weight for importance of item A_i;
W_s: the fuzzy weight for importance of itemset *s*;
$connt_i$: the number of item A_i appearing in the *n* transactions;
α: the predefined linguistic minimum support value;
β: the predefined linguistic minimum confidence value;
I_j: the *j*-th membership function of importance;
I^{ave}: the fuzzy average weight of all possible linguistic terms of importance;
$wsup_i$: the fuzzy weighted support of item A_i;
$wconf_R$: the fuzzy weighted confidence of rule *R*;
minsup: the transformed fuzzy set from the linguistic minimum support value α;
wminsup: the fuzzy weighted set of minimum supports;
minconf: the transformed fuzzy set from the linguistic minimum confidence value β;
wminconf: the fuzzy weighted set of minimum confidences;
C_r: the set of candidate itemsets with *r* items;
L_r: the set of large itemsets with *r* items.

5 Weighted Data Mining for Linguistic Minimum Supports and Minimum Confidences

In this section, the fuzzy concepts are used to represent item importance, minimum supports and minimum confidences. The proposed mining algorithm first uses the set of membership functions for importance to transform managers' linguistic evaluations of item importance into fuzzy weights. The fuzzy weights of an item from different mangers are then averaged. The algorithm then calculates the weighted counts of all items from the transaction data and according to the average fuzzy weights of items. The given linguistic minimum support value is also transformed into a fuzzy weighted set. All weighted large 1-itemsets can thus be found by ranking the fuzzy weighted support of each item with fuzzy weighted minimum support. After that, candidate 2-itemsets are formed from the weighted large 1-itemsets and the same procedure is used to find all weighted large 2-itemsets. This procedure is repeated until all weighted large itemsets have been found. The fuzzy weighted confidences from large itemsets are then calculated to find interesting association rules. Details of the proposed mining algorithm are described below.

The algorithm:

INPUT: A set of n transaction data, a set of m items with their importance evaluated by k managers, three sets of membership functions respectively for importance, minimum support and minimum confidence, a pre-defined linguistic minimum support value α, and a pre-defined linguistic minimum confidence value β.

OUTPUT: A set of weighted association rules.

STEP 1: Transform each linguistic term of importance for item A_i, $1 \le i \le m$, which is evaluated by the j-th manager into a fuzzy set $W_{i,j}$ of weights, using the given membership functions of item importance.

STEP 2: Calculate the fuzzy average weight W_i^{ave} of each item A_i by fuzzy addition as:

$$W_i^{ave} = \frac{1}{k} * \sum_{j=1}^{k} W_{i,j} \; .$$

STEP 3: Calculate the number ($count_i$) of each item A_i appearing in the set of transactions.

STEP 4: Calculate the fuzzy weighted support $wsup_i$ of each item A_i as:

$$wsup_i = \frac{count_i \times W_i^{ave}}{n} \, ,$$

where n is the number of transactions.

STEP 5: Transform the given linguistic minimum support value α into a fuzzy set *minsup* of minimum supports, using the given membership functions of minimum supports.

STEP 6: Calculate the fuzzy weighted set (*wminsup*) of the given minimum support value as:

$$wminsup = minsup \times \text{(the gravity of } I^{ave}\text{)},$$

where

$$I^{ave} = \frac{\sum_{j=1}^{m} I_j}{m} \, ,$$

with I_j being the j-th membership function of importance. I^{ave} represents the fuzzy average weight of all possible linguistic terms of importance.

STEP 7: Check whether the weighted support ($wsup_i$) of each item A_i is larger than or equal to the fuzzy weighted minimum support (*wminsup)* by fuzzy ranking. Any fuzzy ranking approach can be applied here as long as it can generate a crisp rank. If $wsup_i$ is equal to or greater than *wminsup*, put A_i in the set of large 1-itemsets L_1.

STEP 8: Set $r = 1$, where r is used to represent the number of items kept in the current large itemsets.

STEP 9: Generate the candidate set C_{r+1} from L_r in a way similar to that in the *apriori* algorithm [4]. That is, the algorithm first joins L_r and L_r assuming that r-1 items in the two itemsets are the same and the other one is different. It then keeps in C_{r+1} the itemsets, which have all their sub-itemsets of r items existing in L_r.

STEP 10: Do the following substeps for each newly formed (r+1)-itemset s with items ($s_1, s_2, \ldots, s_{r+1}$) in C_{r+1}:

(a) Calculate the fuzzy weight W_s of itemset s as:

$$W_s = W_{s_1}^{ave} \Lambda W_{s_2}^{ave} \Lambda \ldots \Lambda W_{s_{r+1}}^{ave},$$

where $W_{s_i}^{ave}$ is the fuzzy average weight of item s_i, calculated in STEP 2. If the minimum operation is used for the intersection, then:

$$W_s = \underset{i=1}{\overset{r+1}{Min}} W_{s_i}^{ave}$$

(b) Calculate the number ($count_s$) of itemset s appearing in the transaction data.

(c) Calculate the weighted support $wsup_s$ of itemset s as:

$$wsup_s = \frac{W_s \times count_s}{n},$$

where n is the number of transactions.

(d) Check whether the weighted support $wsup_s$ of itemset s is greater than or equal to the fuzzy weighted minimum support *wminsup* by fuzzy ranking. If $wsup_s$ is greater than or equal to *wminsup*, put s in the set of large ($r+1$)-itemsets L_{r+1}.

STEP 11: IF L_{r+1} is null, then do the next step; otherwise, set $r = r+1$ and repeat Steps 9-11.

STEP 12: Transform the given linguistic minimum confidence value β into a fuzzy set *minconf* of minimum confidences, using the given membership functions of minimum confidences.

Step 13: Calculate the fuzzy weighted set (*wminconf*) of the given minimum confidence value as:

$$wminconf = minconf \times \text{ (the gravity of } I^{ave}),$$

where I^{ave} is the same as that calculated in Step 6.

STEP 14: Construct the association rules from each weighted large q-itemset s with items ($s_1, s_2, \ldots, s_q$), $q \geq 2$, using the following substeps:

(a) Form all possible association rules as follows:

$$s_1 \Lambda \ldots \Lambda s_{j-1} \Lambda s_{j+1} \Lambda \ldots \Lambda s_q \rightarrow s_j, j = 1 \text{ to } q.$$

(b) Calculate the weighted confidence value $wconf_R$ of each possible association rule R as:

$$wconf_R = \frac{count_s}{count_{s-s_j}} \times W_s.$$

(c) Check whether the weighted confidence $wconf_R$ of association rule R is greater than or equal to the fuzzy weighted minimum confidence *wminconf* by fuzzy ranking. If $wconf_R$ is greater than or equal to *wminconf*, keep rule R in the interesting rule set.

STEP 15: For each rule R with weighted support $wsup_R$ and weighted confidence

$wconf_R$ in the interesting rule set, find the linguistic minimum support region S_i and the linguistic minimum confidence region C_j with $wminsup_{i-1} \leq wsup_R < wminsup_i$ and $wminconf_{j-1} \leq wconf_R < wminconf_j$ by fuzzy ranking, where:

$$wminsup_i = minsup_i \times \text{ (the gravity of } I^{ave}),$$

$$wminconf_j = minconf_j \times \text{ (the gravity of } I^{ave}),$$

$minsup_i$ is the given membership function for S_i and $minconf_j$ is the given membership function for C_j. Output rule R with linguistic support value S_i and linguistic confidence value C_j.

The rules output after step 15 can serve as meta-knowledge concerning the given transactions.

6 An Example

In this section, an example is given to illustrate the proposed data-mining algorithm. This is a simple example to show how the proposed algorithm can be used to generate weighted association rules from a set of transactions. The data set includes ten transactions, as shown in Table 1.

Table 1: The data set used in this example

TID	Items
1	ABC
2	ACE
3	ABF
4	BCD
5	BCE
6	ABDE
7	ABCE
8	ABCEF
9	ABCEF
10	BCDEF

Each transaction is composed of a transaction identifier and items purchased. There are six items, respectively being *A*, *B*, *C*, *D*, *E* and *F*, to be purchased. The importance of the items is evaluated by three managers as shown in Table 2.

Table 2: The item importance evaluated by three managers

Item	Manager1	Manger2	Manager3
A	Important	Ordinary	Ordinary
B	Very Important	Important	Important
C	Ordinary	Important	Important
D	Unimportant	Unimportant	Very Important
E	Important	Important	Important
F	Unimportant	Unimportant	Ordinary

Assume the membership functions for item importance are given in Figure 2. In Figure 2, item importance is divided into five fuzzy regions: Very Unimportant, Unimportant, Ordinary, Important and Very Important. Each fuzzy region is represented by a membership function. For simplicity, triangular membership functions are used here. The membership functions in Figure 2 can be represented as follows:

Very Unimportant (VU): (0, 0, 0.25),
Unimportant (U): (0, 0.25, 0.5),
Ordinary (O): (0.25, 0.5, 0.75),
Important (I): (05, 075, 1), and
Very Important (VI): (0.75, 1, 1).

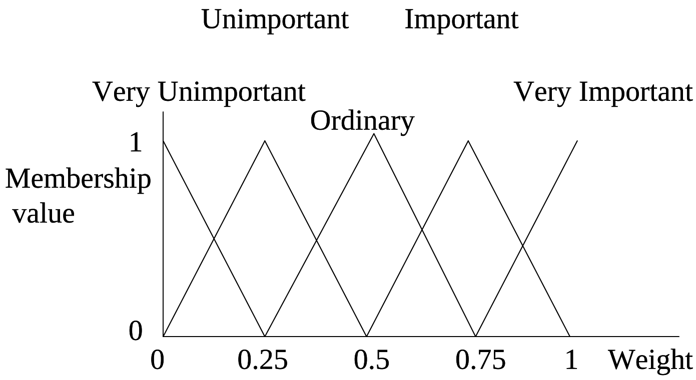


Figure 2: The membership functions of item importance used in this example

For the training data given in Table 1, the proposed mining algorithm proceeds as follow.

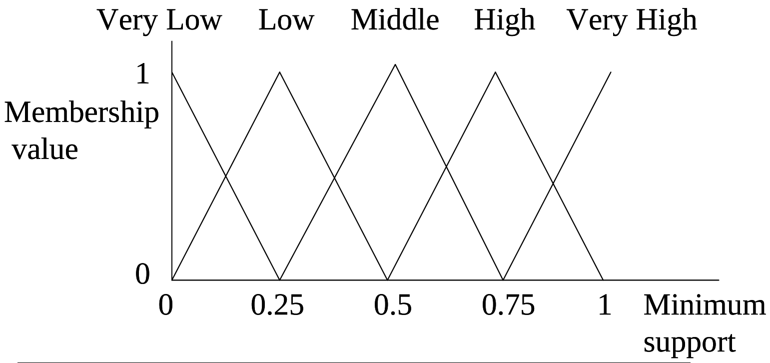
Step 1: The linguistic terms for item importance given in Table 2 are transformed into fuzzy sets by the membership functions in Figure 2. For example, item *A* is evaluated to be important by Manager 1, and can then be transformed as a triangular fuzzy set (0.5, 0.75, 1) of weights. The transformed results for Table 2 are shown in Table 3.

Table 3: The fuzzy weights transformed from the item importance in Table 2.

Item	Manager1	Manger2	Manager3
A	(0.5, 0.75,1)	(0.25, 0.5, 0.75)	(0.25, 0.5, 0.75)
B	(0.75, 1, 1)	(0.5, 0.75,1)	(0.5, 0.75,1)
C	(0.25, 0.5, 0.75)	(0.5, 0.75,1)	(0.5, 0.75,1)
D	(0, 0.25, 0.5)	(0, 0.25, 0.5)	(0.75, 1, 1)
E	(0.5, 0.75,1)	(0.5, 0.75,1)	(0.5, 0.75,1)
F	(0, 0.25, 0.5)	(0, 0.25, 0.5)	(0.25, 0.5, 0.75)

Step 2: The average weight of each item is calculated by fuzzy addition. The results are shown in Table 4.

Table 4: The average fuzzy weights of all the items



Item	Average Fuzzy Weight
A	(0.333, 0.583, 0.833)
B	(0.583, 0.833, 1)
C	(0.417, 0.667, 0.917)
D	(0, 0.167, 0.417)
E	(0.5, 0.75, 1)
F	(0.083, 0.333, 0.583)

Step 3: The appearing number (count) of each item is counted from the transactions in Table 1. Results for all the items are shown in Table 5.

Table 5: The counts of all the items

Item	Count
A	7
B	9
C	8
D	3
E	7
F	4

Step 4: The weighted support of each item is calculated. Results for all the items are shown in Table 6.

Table 6: The weighted supports of all the items

Item	Weighted Support
A	(0.233, 0.408, 0.583)
B	(0.525, 0.75, 0.9)
C	(0.333, 0.533, 0.733)
D	(0, 0.05, 0.125)
E	(0.35, 0.525, 0.7)
F	(0.033, 0.133, 0.233)

Step 5: The given linguistic minimum support value is transformed into a fuzzy set of minimum supports. Assume the membership functions for minimum supports are given in Figure 3.

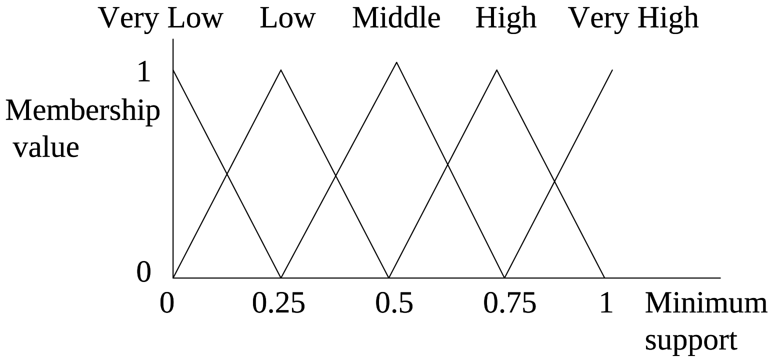


Figure 3: The membership functions of minimum supports

Also assume the given linguistic minimum support value is “High”. It is then transformed into a fuzzy set of minimum supports, (0.5, 0.75, 1), according to the given membership functions in Figure 3.

Step 6: The fuzzy average weight of all possible linguistic terms of importance in Figure 3 is first calculated as:

$$I^{ave} = [(0, 0, 0.25) + (0, 0.25, 0.5) + (0.25, 0.5, 0.75) + (0.5, 0.75, 1) +(0.75, 1, 1)]/5 = (0.3, 0.5, 0.7).$$

The gravity of I^{ave} is then (0.3+0.5+0.7)/3, which is 0.5. The fuzzy weighted set of minimum supports for “High” is then (0.5, 0.75, 1) × 0.5, which is (0.25, 0.375, 0.5).

Step 7: The weighted support of each item is compared with the fuzzy weighted minimum support by fuzzy ranking. Assume the gravity ranking approach is adopted in this example. Take Item *A* as an example. The average height of the weighted support for item *A* is (0.233 + 0.408 + 0.583)/3, which is 0.408. The average height of the fuzzy weighted minimum support is (0.25 + 0.375 + 0.5)/3, which is 0.375. Since 0.408 > 0.375, *A* is thus a large weighted 1-itemset. Similarly, *B*, *C* and *E* are large weighted 1-itemsets. These 1-itemsets are put in L_1.

Step 8: *r* is set at 1, where *r* is used to store the number of items kept in the current itemsets.

Step 9: The candidate set C_{r+1} is generated from L_r. C_2 is then first generated from L_1 as follows: (*A, B*), (*A, C*), (*A, E*), (*B, C*), (*B, E*) and (*C, E*).

Step 10: The following substeps are done for each newly formed candidate itemset in C_2.

(a) The fuzzy weights of all the itemsets in C_2 are calculated. Here, the minimum operator is used for intersection. Take Itemset (A, B) as an example. Its fuzzy weighted support value is calculated as *min*((0.333, 0.583, 0.833), (0.583, 0.833, 1)), which is (0.333, 0.583, 0.833). The results for all the 2-itemsets are shown in Table 7.

Table 7: The fuzzy weights of the 2-itemsets in C_2.

Itemset	Fuzzy Weight
(A, B)	(0.333, 0.583, 0.833)
(A, C)	(0.333, 0.583, 0.833)
(A, E)	(0.333, 0.583, 0.833)
(B, C)	(0.417, 0.667, 0.917)
(B, E)	(0.5, 0.75, 1)
(C, E)	(0.417, 0.667, 0.917)

(b) The scalar cardinality (count) of each candidate 2-itemset is counted from the transaction data. Results for this example are shown in Table 8.

Table 8: The counts of the itemsets in C_2.

Itemset	Count
(A, B)	6
(A, C)	5
(A, E)	5
(B, C)	7
(B, E)	5
(C, E)	6

(c) The weighted support of each candidate 2-itemset is calculated. Take (*A*, *B*) as an example. The minimum fuzzy weight of (*A, B*) is (0.333, 0.583, 0.833) and the count is 6. Its weighted support is then (0.333, 0.583, 0.833)*6/10, which is (0.2, 0.35, 0.5). All the weighted supports of the candidate 2-itemsets are shown in Table 9.

Table 9: The weighted supports of the 2-items

Itemset	Fuzzy Weight
(A, B)	(0.2, 0.35, 0.5)
(A, C)	(0.167, 0.292, 0.417)
(A, E)	(0.167, 0.292, 0.417)
(B, C)	(0.292, 0.467, 0.642)
(B, E)	(0.25, 0.375, 0.5)
(C, E)	(0.25, 0.4, 0.55)

(d) The weighted support of each candidate 2-itemset is compared with the fuzzy weighted minimum support by fuzzy ranking. As mentioned above, the gravity ranking approach is adopted in this example. (*B*, *C*), (*B*, *E*) and (*C*, *E*) are then found to be large weighted 2-itemsets. These 2-itemsets are put in L_2.

Step 11: Since L_2 is not null in the example, $r = r + 1 = 2$. Steps 9 to 11 are repeated to find L_3. C_3 is then generated from L_2. Only the 3-itemset (*B*, *C*, *E*) is formed in this example. The average height of its fuzzy weighted support is 0.34, smaller than 0.375. It is not put in L_3, and L_3 is thus an empty set.

Step 12: The given linguistic minimum confidence value is transformed into a fuzzy set of minimum confidences. Assume the membership functions for minimum confidence values are shown in Figure 4, which are similar to those in Figure 3.

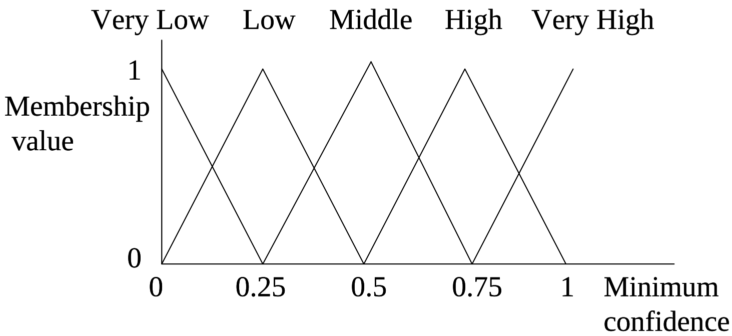


Figure 4: The membership functions of minimum confidences

Also assume the given linguistic minimum confidence value is “High”. It is then transformed into a fuzzy set of minimum confidences, (0.5, 0.75, 1), according to the given membership functions in Figure 4.

Step 13: The fuzzy average weight of all possible linguistic terms of importance is the same as that found in Step 5. Its gravity is thus 0.5. The fuzzy weighted set of minimum confidences for “High” is then (0.5, 0.75, 1) $\times$ 0.5, which is (0.25, 0.375, 0.5).

Step 14: The association rules from each large itemset are constructed by using the following substeps.

(a) All possible association rules are formed as follows:

If *B*, then *C*;
If *C*, then *B*;
If *B*, then *E*;
If *E*, then *B*;
If *C*, then *E*;
If *E*, then *C*.

(b) The weighted confidence factors for the above possible association rules are calculated. Take the first possible association rule as an example. The counts of B and $B \cap C$ are respectively 0.9 and 0.7, and the fuzzy weight of $B \cap C$ is (0.417, 0.667, 0.917). The weighted confidence factor for the association rule “*If B, then C*” is then:

$$\frac{0.7}{0.9} \times (0.417, 0.667, 0.917) = (0.324, 0.519, 0.713).$$

(c) The weighted confidence of each association rule is compared with the fuzzy weighted minimum confidence by fuzzy ranking. A. In this example, the following six rules are put in the interesting rule set:

1. If *B* is bought then *C* is bought;
2. If *C* is bought then *B* is bought;
3. If *B* is bought then *E* is bought;
4. If *E* is bought then *B* is bought;
5. If *C* is bought then *E* is bought;

6. If *E* is bought then *C* is bought.

STEP 15: The linguistic support and confidence values are found for each rule *R*. Take the third interesting association rule "*If B, then E*" as an example. Its weighted support is (0.25, 0.375, 0.5) and weighted confidence is (0.278, 0.417, 0.556). Since the membership function for linguistic minimum support region "High" is (0.5, 0.75, 1) and for "Very High" is (0.75, 1, 1). The weighted fuzzy set for these two regions are (0.25, 0.375, 0.5) and (0.375, 0.5, 0.5). Since (0.25, 0.375, 0.5) ≤ (0.25, 0.375, 0.5) < (0.375, 0.5, 0.5) by fuzzy ranking, the linguistic support value for Rule *R* is then High. Similarly, the linguistic confidence value for Rule *R* is High. All the six interesting association rules are then output as:

1. If *B* is bought then *C* is bought, with a very high support and a very high confidence;
2. If *C* is bought then *B* is bought, with a very high support and a very high confidence;
3. If *B* is bought then *E* is bought, with a high support and a high confidence;
4. If *E* is bought then *B* is bought, with a high support and a very high confidence;
5. If *C* is bought then *E* is bought, with a very high support and a very high confidence;
6. If *E* is bought then *C* is bought, with a very high support and a very high confidence.

The six rules above are thus output as meta-knowledge concerning the given weighted transactions.

7 Conclusion and Future Work

In this paper, we have proposed a new weighted data-mining algorithm to find interesting association rules with linguistic supports and confidences. Items are evaluated by managers as linguistic terms, which are then transformed and averaged as fuzzy sets of weights. Fuzzy operations including fuzzy ranking are used to find weighted large itemsets and association rules. Compared to previous mining approaches, the proposed one has linguistic inputs and outputs, which are more natural and understandable for human beings.

Acknowledgement.
This research was supported by the National Science Council of the Republic of China under contract NSC94-2213-E-390-005.

References

[1] R. Agrawal, T. Imielinksi and A. Swami, Mining association rules between sets of items in large database, The 1993 ACM SIGMOD Conference, Washington DC, USA, 1993.

[2] R. Agrawal, T. Imielinksi and A. Swami, Database mining: a performance perspective, IEEE Transactions on Knowledge and Data Engineering, Vol. 5, No. 6, 1993, 914-925.

[3] R. Agrawal, R. Srikant and Q. Vu, Mining association rules with item constraints, The Third International Conference on Knowledge Discovery in Databases and Data Mining, Newport Beach, California, August 1997.
[4] R. Agrawal and R. Srikant, Fast algorithm for mining association rules, The International Conference on Very Large Databases, 1994, 487-499.
[5] A. F. Blishun, Fuzzy learning models in expert systems, Fuzzy Sets and Systems, Vol. 22, 1987, 57-70.
[6] C. H. Cai, W. C. Fu, C. H. Cheng and W. W. Kwong, Mining association rules with weighted items, The International Database Engineering and Applications Symposium, 1998, 68-77.
[7] L. M. de Campos and S. Moral, Learning rules for a fuzzy inference model, Fuzzy Sets and Systems, Vol. 59, 1993, 247-257.
[8] K. C. C. Chan and W. H. Au, Mining fuzzy association rules, The 6th ACM International Conference on Information and Knowledge Management, 1997, 209-215.
[9] R. L. P. Chang and T. Pavliddis, Fuzzy decision tree algorithms, IEEE Transactions on Systems, Man and Cybernetics, Vol. 7, 1977, 28-35.
[10] M. S. Chen, J. Han and P. S. Yu, Data mining: an overview from a database perspective, IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No.6, 1996, 866-883.
[11] C. Clair, C. Liu and N. Pissinou, Attribute weighting: a method of applying domain knowledge in the decision tree process, The Seventh International Conference on Information and Knowledge Management, 1998, 259-266.
[12] P. Clark and T. Niblett, The CN2 induction algorithm, Machine Learning, Vol. 3, 1989, 261-283.
[13] M. Delgado and A. Gonzalez, An inductive learning procedure to identify fuzzy systems, Fuzzy Sets and Systems, Vol. 55, 1993, 121-132.
[14] A. Famili, W. M. Shen, R. Weber and E. Simoudis, Data preprocessing and intelligent data analysis, Intelligent Data Analysis, Vol. 1, No. 1, 1997.
[15] W. J. Frawley, G. Piatetsky-Shapiro and C. J. Matheus, Knowledge discovery in databases: an overview, The AAAI Workshop on Knowledge Discovery in Databases, 1991, 1-27.
[16] A.Gonzalez, A learning methodology in uncertain and imprecise environments, International Journal of Intelligent Systems, Vol. 10, 1995, 57-371.
[17] I. Graham and P. L. Jones, Expert Systems – Knowledge, Uncertainty and Decision, Chapman and Computing, Boston, 1988, 117-158.
[18] T. P. Hong and J. B. Chen, Finding relevant attributes and membership functions, Fuzzy Sets and Systems, Vol.103, No. 3, 1999, 389-404.
[19] T. P. Hong and J. B. Chen, Processing individual fuzzy attributes for fuzzy rule induction, accepted and to appear in Fuzzy Sets and Systems.
[20] T. P. Hong and C. Y. Lee, Induction of fuzzy rules and membership functions from training examples, Fuzzy Sets and Systems, Vol. 84, 1996, pp. 33-47.
[21] T. P. Hong and S. S. Tseng, A generalized version space learning algorithm for noisy and uncertain data, IEEE Transactions on Knowledge and Data Engineering, Vol. 9, No. 2, 1997, 336-340.
[22] T. P. Hong, C. S. Kuo and S. C. Chi, Mining association rules from quantitative data, Intelligent Data Analysis, Vol. 3, No. 5, 1999, 363-376.
[23] A. Kandel, Fuzzy Expert Systems, CRC Press, Boca Raton, 1992, 8-19.

[24] C. M. Kuok, A. W. C. Fu and M. H. Wong, Mining fuzzy association rules in databases, The ACM SIGMOD Record, Vol. 27, No. 1, 1998, 41-46.

[25] E. H. Mamdani, Applications of fuzzy algorithms for control of simple dynamic plants, IEEE Proceedings, 1974, 1585-1588.

[26] H. Mannila, Methods and problems in data mining, The International Conference on Database Theory, 1997.

[27] J. R. Quinlan, Decision tree as probabilistic classifier, The Fourth International Machine Learning Workshop, Morgan Kaufmann, San Mateo, CA, 1987, 31-37.

[28] J. R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA, 1993.

[29] J. Rives, FID3: fuzzy induction decision tree, The First International symposium on Uncertainty, Modeling and Analysis, 1990, 457-462.

[30] C. H. Wang, T. P. Hong and S. S. Tseng, Inductive learning from fuzzy examples, The fifth IEEE International Conference on Fuzzy Systems, New Orleans, 1996, 13-18.

[31] C. H. Wang, J. F. Liu, T. P. Hong and S. S. Tseng, A fuzzy inductive learning strategy for modular rules, Fuzzy Sets and Systems, Vol. 103, No. 1, 1999, 91-105.

[32] R.Weber, Fuzzy-ID3: a class of methods for automatic knowledge acquisition, The Second International Conference on Fuzzy Logic and Neural Networks, Iizuka, Japan, 1992, 265-268.

[33] Y. Yuan and M. J. Shaw, Induction of fuzzy decision trees, Fuzzy Sets and Systems, Vol. 69, 1995, 125-139.

[34] S. Yue, E. Tsang, D. Yeung and D. Shi, Mining fuzzy association rules with weighted items, The IEEE International Conference on Systems, Man and Cybernetics, 2000, 1906-1911.

[35] L. A. Zadeh, Fuzzy logic, IEEE Computer, 1988, 83-93.

[36] L. A. Zadeh, Fuzzy sets, Information and Control, Vol. 8, No. 3, 1965, 338-353.

Received: April, 2012